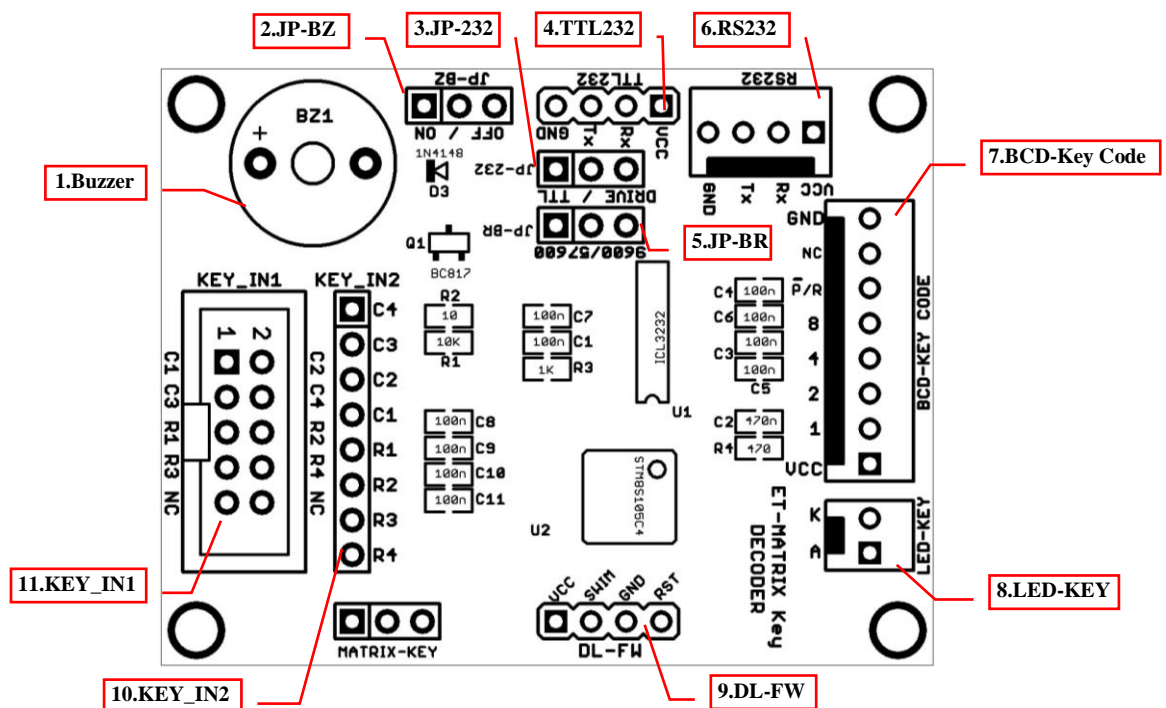


ET-MATRIX KEY DECODER เป็น Module สำหรับถอดรหัสการกด Key Pad แบบ Matrix 4x4 และ 4x3 ซึ่งจะส่งค่า Key ที่ถูกกดออกมาให้ผู้ใช้ 2 แบบ คือ แบบ BCD 8421 พร้อมสถานะการกดปล่อย Key ที่ขั้วต่อ Connector 8 Pin และแบบ ASCII Code ผ่านทางขั้วต่อ RS232 ตัว Module จะมี Buzzer และ Connector สำหรับต่อ LED เพื่อแสดงสถานะ การ กด/ปล่อย Key โดย Key Pad ที่จะนำมาต่อใช้งานนั้น จะต้องมีการจัดเรียง Pin ของ Matrix Key ตรงกับขั้วต่อ Key_IN ของ Module ด้วย

1.คุณสมบัติ Module ET-Matrix Key Decoder

- รองรับ Key Pad แบบ Matrix 4x4 และ 4x3 โดยลักษณะ Key Pad ที่ต่อได้โดยตรงสามารถดูได้ในหัวข้อ โครงสร้าง Module ส่วน Key Pad ที่มีการจัดเรียง Pin ไม่ตรงกับขั้วต่อ Key_IN ของ Module ผู้ใช้สามารถจัดเรียง Pin เชื่อมต่อสาย Key Pad เองใหม่ได้
- ตัว Module มี Connector รองรับ Key Pad 2 Key_In คือแบบ Block 10 Pin และแบบแถวเดียว 8 Pin ซึ่งแต่ละ Key_In จะมีการจัดเรียง Pin Matrix ต่างกัน ดังนั้นผู้ใช้จะต้องเลือกขั้วต่อ Key_IN ให้สอดคล้องกับ โครงสร้างของ Key Pad ที่จะนำมาต่อใช้งาน
- ไฟเลี้ยง Module +3.3V หรือ +5V
- แสดงสถานะ การกด Key ด้วยเสียง Beep ของ Buzzer (Set On/Off ได้ด้วย Jumper) และมี Connector สำหรับใช้ต่อ LED ให้ 1 ช่อง เพื่อใช้แสดงสถานะ การกด Key เช่นกัน
- ตัว Module จะแสดงสถานะ เริ่มต้นทำงาน(Power On) ด้วยเสียง Beep 2 ครั้ง
- การทำงานของ Module จะทำงานแบบ Single Key คือสามารถกดใช้งานได้ครั้งละ Key ไม่สามารถกดพร้อมกัน 2 Key ได้
- Output Key Code ของ Key ที่กด จะให้ค่าออกมา 2 แบบ คือ
 - 1) แบบ Binary Code (BCD8421) จะส่งค่า Key Code ออกมาที่ Connector 8 Pin โดยมี Pin \bar{P}/R บอกสถานะ การกด/ปล่อย Key
 - 2)แบบ ASCII Code จะส่งค่า Key Code ผ่านทางขั้วต่อ RS232(แบบ TTL และ Line Drive) ซึ่งสามารถกำหนด Baud Rate ใช้งานได้ 2 ค่าคือ 9600 กับ 57600 bit/s โดยค่า Key Code ที่ส่งออกมานี้จะนำด้วยอักษร 'P' หรือ 'R' เพื่อบอกสถานะ การ กด/ปล่อย Key แล้วจึงตามด้วยค่า ASCII ของ Key ที่กด

2.โครงสร้าง Module ET-Matrix Key Decoder

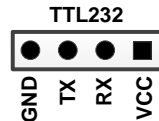


- 1. Buzzer : เป็นตัวกำเนิดเสียง Beep เมื่อ Module Power-ON และเมื่อ Key ถูกกด
- 2. JP-BZ : เป็น Jumper สำหรับ ON/OFF Buzzer จะมีผลทันทีเมื่อมีการเปลี่ยนแปลงการ Set
- 3. JP-232: เป็น Jumper สำหรับเลือกขั้วต่อใช้งาน 232 โดยถ้า Set มาทางด้าน TTL จะเป็นการเลือกใช้งานขั้วต่อ TTL232(หมายเลข4) แต่ถ้า

Set Jumper ไปทางด้าน Drive จะเป็นการเลือกใช้งานขั้วต่อ RS232(หมายเลข6)

* ในการอ่าน Key Code แบบ ASCII Code ผู้ใช้ต้องเลือกใช้งานขั้วต่อ 232 ขั้วต่อใดขั้วต่อหนึ่งเท่านั้น และ Set Jumper JP-232 ให้ถูกต้องกับขั้วต่อที่จะใช้งานด้วย ซึ่งการจะเลือกใช้งานขั้วต่อใดจะขึ้นอยู่กับบอร์ด MCU ที่ผู้ใช้นำมาต่ออ่านค่า Key Code โดยถ้าบอร์ดที่นำมาต่อมีการต่อ IC Line Driver 232 ไว้แล้ว หรือ ต้องการต่อไปยัง Port 232 ของ PC ก็ให้เลือกใช้ขั้วต่อ RS232(หมายเลข6) และ Set Jumper JP-232 มาทางด้าน Drive แต่ถ้าจะนำไปต่อกับขา Uart ของ MCU โดยตรง ก็ให้เลือกใช้งานขั้วต่อ TTL232 (หมายเลข4) และ Set Jumper JP-232 มาทางด้าน TTL เป็นต้น

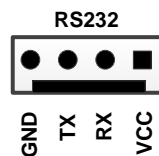
-4. TTL232: เป็นขั้วต่อ 232 ที่มีระดับสัญญาณแบบ TTL ซึ่งขั้วต่อนี้ผู้ใช้สามารถต่อเข้ากับ Pin Uart (Rx,Tx) ของ MCU ได้โดยตรง ในการต่อให้ขั้วสายระหว่าง Rx,Tx ด้วย ซึ่งมีการจัดเรียงขาตั้งรูป โดย Pin VCC และ GND จะใช้ต่อไฟเลี้ยง 3.3V หรือ 5V ให้กับ Module



รูปแสดงการจัดเรียง Pin ของขั้วต่อ TTL232

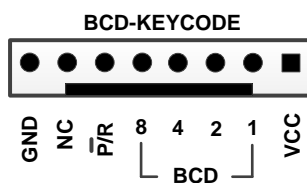
-5. JP-BR : เป็น Jumper กำหนด Baud Rate ให้กับการสื่อสารข้อมูลทาง 232 โดยสามารถเลือกได้ 2 ค่า คือ 9600 และ 57600 bit/s เมื่อต้องการใช้งานค่า Baud Rate เท่าไหร่ ก็ให้ Set Jumper ไปทางด้านนั้น โดยการ Set Baud Rate จะมีผลหลังจาก Power On Module ใหม่

-6. RS232 : เป็นขั้วต่อ RS232 ที่มี IC Line Driver ต่อไว้ให้บนบอร์ดแล้ว ดังนั้นในการใช้งานผู้ใช้สามารถต่อเข้ากับ Port 232 ของ PC หรือ Port 232 ของ MCU ที่มีการต่อ IC Line Driver ไว้ให้เช่นเดียวกันได้ ในการต่อให้ขั้วสายระหว่าง Rx,Tx ด้วย โดยมีการจัดเรียงขาตั้งรูปด้านล่าง ซึ่ง Pin VCC และ GND จะใช้ต่อไฟเลี้ยง 3.3V หรือ 5V ให้กับ Module



รูปแสดงการจัดเรียง Pin ของขั้วต่อ RS232

-7. BCD-Key Code : เป็น Connector ขนาด 8 Pin ทำหน้าที่ในการส่งค่า Key Code ของ Key ที่กดหรือปล่อย ในรูปแบบของ Binary BCD8421 รวมทั้งส่งค่าสถานะ การกด/ปล่อย Key ออกมาให้ผู้ใช้ โดยมีการจัดเรียง Pin ดังรูป



รูปแสดงการจัดเรียง Pin Connector BCD-KEY CODE

รายละเอียดของ Pin

VCC/GND = ใช้ต่อไฟเลี้ยง Module 3.3V (เมื่อใช้กับ MCU3.3V) หรือ 5V (เมื่อใช้กับ MCU 5V)

BCD = เป็น Pin สำหรับส่งค่า Key Code 4 Bit โดย PIN หมายเลข 8 เป็นบิต MSB ค่า Key Code จะถูก ส่งออกมาทุกครั้งที่มีการกด Key และปล่อย Key โดยค่า Key Code ที่ส่งออกมา จะเป็นค่าของ Key ที่กดหรือปล่อยล่าสุด และค้างค่านั้นไว้จนกว่า จะมีการกด Key อื่นๆใหม่

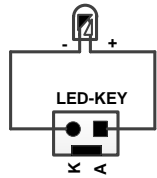
P/R = เป็น Pin สำหรับส่งค่าสถานะ การ กด/ปล่อย Key โดยในสภาวะปกติไม่มีการกด Key , Pin นี้จะให้สถานะ Logic ออกมาเป็น 1 ค้างไว้ เมื่อมีการกด Key , Pin นี้จะให้ Logic ออกมาเป็น 0 ค้างไว้จนกว่าจะมีการ ปล่อย Key

NC = No Connect ไม่ต่อใช้งาน

Note ในการอ่านค่า Key Code นั้นผู้ใช้ควรเลือกอ่านค่าแบบใดแบบหนึ่งเท่านั้น ระหว่าง ASCII Code หรือ Binary Code เพื่อเป็นการเลือกต่อใช้งาน Connector ชุดใดชุดหนึ่งเท่านั้น โดยถ้าต้องการอ่านค่า Key Codec แบบ ASCII Code ก็ให้เลือกต่อ ใช้งาน Connector RS232หรือ TTL232 แต่ถ้าต้องการอ่านค่า Key Code แบบ Binary BCD8421 ก็ให้เลือกต่อ ใช้งาน Connector BCD-Key Code

ในส่วนของไฟเลี้ยงบอร์ดนั้น(VCC,GND) ผู้ใช้สามารถเลือกต่อเข้าที่ Connector ตัวใดตัวหนึ่งเท่านั้นระหว่าง Connector RS232 หรือ TTL232 หรือ BCD-Key Code

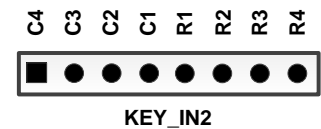
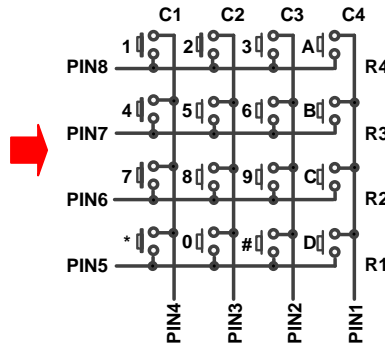
-8. LED-KEY : เป็นขั้วต่อ 2 Pin ใช้สำหรับต่อ LED เพื่อใช้แสดงสถานะ การ กด Key โดยให้ต่อขั้วบวกของ LED เข้ากับ Pin A และ ขั้วลบให้ต่อเข้ากับ Pin K เมื่อมีการกด Key จะเห็น LED ดิด และเมื่อมีการปล่อย Key LED จะดับ



รูปแสดงการต่อ LED เพื่อใช้แสดงสถานะ การกด Key

-9. DL-FW : ใช้ Upgrade Firmware ให้กับ Module (ปกติจะต้องส่งมาให้ทาง ETT Upgrade)

-10. KEY_IN2 : เป็นขั้วต่อ Key Pad 4x4 แบบแถวเดียว 8 Pin ซึ่งจะออกแบบมาให้ใช้งานกับ Key Pad ที่มีการจัดเรียง Pin และ โครงสร้าง Matrix Key (Row/Column) ที่ตรงกับ Diagram ด้านล่างจึงจะสามารถนำมาต่อเข้ากับขั้วต่อนี้ได้ทันทีโดยไม่ต้องทำการบัดกรีสายใหม่



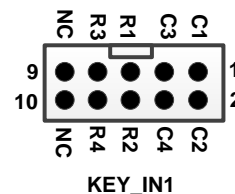
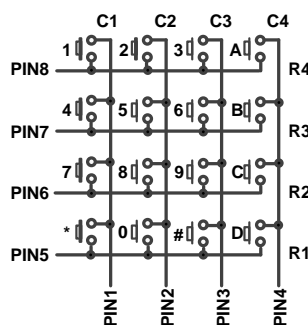
รูปแสดงการจัดเรียง Pin ของขั้วต่อ KEY_IN2

ตัวอย่าง Key Pad 4x4 ที่ใช้งาน

Diagram Key Pad 4x4

สำหรับ Key Pad อื่นๆ ที่มีโครงสร้าง การจัดเรียง Pin ต่างไปจาก Diagram ด้านบน ผู้ใช้สามารถเชื่อมสายเองจาก Key Pad มายังขั้วต่อ KEY_IN2 ได้เช่นกัน โดยเวลาต่อให้อ้างอิงตำแหน่ง Pin Row/Column ตามตัวอย่าง Diagram ด้านบนเป็นหลัก

-11. KEY_IN1 : เป็นขั้วต่อ Key Pad 4x4 แบบบล็อก 10 Pin ซึ่งจะออกแบบมาให้ใช้งานกับ Key Pad ที่มีการจัดเรียง Pin และ โครงสร้าง Matrix Key (Row/Column) ที่ตรงกับ Diagram ด้านล่างจึงจะสามารถนำมาต่อเข้ากับขั้วต่อนี้ได้ทันทีโดยไม่ต้องทำการบัดกรีสายใหม่



รูปแสดงการจัดเรียง Pin ของขั้วต่อ KEY_IN1

ตัวอย่าง Key Pad 4x4 ที่ใช้งาน

Diagram Key Pad 4x4

ส่วน Key Pad 4x3 ที่มีโครงสร้าง Matrix Key และการจัดเรียง Pin และตำแหน่ง Row/Column ที่ตรงกับ Diagram ด้านล่างก็สามารถนำมาต่อเข้ากับขั้วต่อ KEY_IN1 ได้เช่นกัน โดยจะต้องใช้ ET-Convert Key M1 สำหรับแปลง Port Key เป็นแบบ Blok 10 Pin เสียก่อน



ลักษณะ Key Pad 4x3

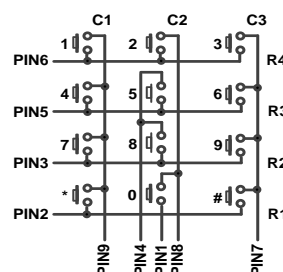


Diagram Key Pad 4x3

สำหรับ Key Pad 4x4 และ 4x3 ที่มีโครงสร้าง การจัดเรียง Pin ต่างไปจาก Diagram ด้านบน ผู้ใช้สามารถเชื่อมต่อสายเองจาก Key Pad มายังขั้วต่อ KEY_IN1 ได้เช่นกัน โดยให้อ้างอิงตำแหน่ง Pin Row/Column ตามตัวอย่าง Diagram ของ Key_IN1 ซึ่งในการต่อสายจาก KEY PAD แบบ 4x3 มายัง KEY_IN1 นั้น ด้าน Row ของ Key PAD 4x3 ให้ต่อกับ Pin R1,R2,R3,R4 และด้าน Column ให้ต่อกับ Pin C1,C2,C3 ของ KEY_IN1

3. การทำงานและการอ่านค่า Key Code ของ ET-MATRIX KEY DECODER

การทำงาน เมื่อจ่ายไฟให้กับ Module จะมีเสียง Beep ดังขึ้น 2 ครั้ง , ที่ขั้วต่อ BCD-KEYCOD ค่าสถานะของ PIN BCD จะเป็น Logic0 และ Pin P/R จะเป็น Logic1 ส่วนที่ขั้ว 232 จะไม่มีการส่ง data ใดๆออกมา และเมื่อมีการกด Key จะมีเสียง Beep 1 ครั้ง ที่ Connector LED-Key ถ้ามีการต่อ LED ไว้ LED ก็จะติดค้างจนกว่าจะมีการปล่อย Key ที่กดอยู่ LED จึงจะดับ

ทุกครั้งที่มีการกดหรือปล่อย Key Output Key Code ทั้งแบบ Binary และ ASCII จะถูก ส่งออกมาที่ Connector 232 และ BCD-Key Code เสมอ ส่วนสถานะของสัญญาณ P/R จะให้ Logic 0 เมื่อ Key ถูกกด และจะกลับมาเป็น Logic 1 เมื่อมีการปล่อย Key

ตาราง KEY CODE ของ ET-MATRIX KEY DECODER

KEY	FOR Binary MODE					FOR ASCII Mode (RS232 TTL-Line Drive)	
	BCD 8421 KEY CODE					ASCII KEY CODE	
	8	4	2	1	HEX	ASCII	HEX
1	0	0	0	1	0x01	'1'	0x31
2	0	0	1	0	0x02	'2'	0x32
3	0	0	1	1	0x03	'3'	0x33
4	0	1	0	0	0x04	'4'	0x34
5	0	1	0	1	0x05	'5'	0x35
6	0	1	1	0	0x06	'6'	0x36
7	0	1	1	1	0x07	'7'	0x37
8	1	0	0	0	0x08	'8'	0x38
9	1	0	0	1	0x09	'9'	0x39
0	0	0	0	0	0x00	'0'	0x30
A	1	0	1	0	0x0A	'A'	0x41
B	1	0	1	1	0x0B	'B'	0x42
C	1	1	0	0	0x0C	'C'	0x43
D	1	1	0	1	0x0D	'D'	0x44
E(#)	1	1	1	0	0x0E	'E'	0x45
F(*)	1	1	1	1	0x0F	'F'	0x46

Note สำหรับ Matrix Key 4x3 ค่า Code จะเป็นไปตาม Key 0-9 และ E-F โดย Key E=Key # และ Key F = Key *

3.1 การอ่าน Key Code แบบ Binary BCD8421

การอ่านค่า Key Code แบบ Binary BCD8421 จะอ่านค่าจาก Connector BCD-Key Code โดยสัญญาณต่างๆของ Connector จะมีการเปลี่ยนแปลงตามสถานะการทำงานดังนี้

- *สถานะ เริ่มต้น(Default)* = เมื่อ Power-ON ที่ Connector O/P BCD Key Code จะถูก Set ดังนี้

$$\text{PIN } \bar{P}/R = 1$$

$$\text{PIN8} = 0 ; \text{PIN4} = 0 ; \text{PIN2} = 0 ; \text{PIN1} = 0$$

- *สถานะ กด Key (Press)* = เมื่อผู้ใช้กด Key เสียง Beep จะดังขึ้น LED Status จะติด(ถ้าต่อ LED) และที่ Connector BCD-KEY CODE ก็จะมีการเปลี่ยนแปลงค่าสถานะต่างๆดังนี้

- 1) PIN \bar{P}/R (สัญญาณ กด/ปล่อย Key) จะเปลี่ยนสถานะ logic จาก 1 ไปเป็น 0 ค้างไว้ตลอดที่มีการกด Key
- 2) PIN BCD 8421 จะมีการเปลี่ยนสถานะตามค่า Key Code ของ key ที่กดล่าสุดออกมาค้างไว้ อ้างอิงกับตาราง Key Code ด้านบนซีกซ้ายมือในช่อง HEX

- *สถานะ ปล่อย Key (Release)* = เมื่อผู้ใช้ปล่อย Key ที่ Connector BCD-KEY CODE จะมีการเปลี่ยนแปลงค่าสถานะดังนี้

- 1) PIN \bar{P}/R (สัญญาณ กด/ปล่อย Key) จะเปลี่ยนสถานะ logic จาก 0 ไปเป็น 1 ค้างไว้ตลอดที่ยังไม่มีการกด Key
- 2) PIN BCD8421 จะมีการเปลี่ยนสถานะตามค่า Key Code ของ Key ที่ปล่อยล่าสุดออกมาค้างไว้

3.2 การอ่าน Key Code แบบ ASCII

การอ่าน Key Code แบบ ASCII ผู้ใช้สามารถอ่านค่าได้จาก Connector RS232 หรือ TTL232 ซึ่งจะเป็นการเชื่อมต่อแบบ Serial Port โดย Baud Rate ในการสื่อสาร สามารถกำหนดได้ 2 ค่าคือ 9600 และ 57600 bit/s ด้วย Jumper JP-BR

การเลือกต่อ Connector ใช้งานก็จะขึ้นอยู่กับผู้ใช้ว่า บอร์ด MCU ที่นำมาต่อเข้ากับ Module นั้น มีส่วนใช้งาน Port 232 เป็นแบบ TTL หรือ แบบผ่าน IC Drive 232 ถ้าเป็นแบบ TTL(Rx,Tx มาจากขา MCU โดย ตรง) ก็ให้เลือกต่อเข้าที่ Connector TTL232 และ Set Jumper JP- 232 มาทางด้าน TTL แต่ถ้าเป็นแบบต่อผ่าน IC Drive 232 ออกมา หรือเป็นการต่อไปยัง RS232 ของ PC ก็ ให้ผู้ใช้เลือกต่อเข้าที่ ConnectOr RS232 ของ Module และให้ Set Jumper JP-232 มาทางด้าน DRIVE โดยไม่ว่าจะเลือกต่อใช้งาน Connector ใดก็ตามเวลาต่อผู้ใช้จะต้องต่อ ขา TX เข้ากับขา Rx ของอีกฝั่ง , และต่อขา RX เข้ากับขา Tx ของอีกฝั่ง ซึ่งจะเป็นลักษณะการต่อแบบไขว้สายกัน

สำหรับค่า Key Code ที่ส่งออกมาให้ผู้ใช้ใช้นั้นจะเป็นค่า ASCII Code โดยรูปแบบของค่าที่ส่งออกมานั้นจะเป็นไปตามสถานะการกด Key ดังนี้

- *สถานะเริ่มต้น(Default)* = เมื่อ Power-ON จะไม่มีการส่งข้อมูลใดๆ ออกมาที่ Port 232 นั่นคือถ้าไม่มีการกด Key ตัว Module จะไม่มีการส่งค่าใดๆออกมาทาง Port 232

- *สถานะกด Key (Press)* = เมื่อมีการกด Key Module ก็จะส่ง Data ออกมาที่ Port 232 จำนวน 3 Byte คือ ASCII 'P' จะถูกส่งออกมาเป็น Byte แรก เพื่อบอกสถานะ การกด Key และ ASCII Code ของ Key ที่กดจะถูกส่งออกมาเป็น Byte ที่ 2 และจบด้วยการส่ง 0x0D เป็น Byte สุดท้าย โดยค่า Key Code จะเป็นไปตาม ตาราง Key Code ด้านบนซีกขวามือในช่อง ASCII หรือ HEX

	ASCII CODE		Hex
	Status Key (Byte 1)	Key Code (Byte 2)	End Byte (Byte 3)
PRESS KEY	'P' (0x50)	'0-9', 'A-F'	0x0D

Ex. กด Key F (Key *)ค่าที่ได้ก็จะเป็น PF และ 0x0D

ตารางรูปแบบการส่ง data เมื่อมีการกด Key

-สถานะ ปล่อย Key (Release) = เมื่อผู้ใช้ปล่อย Key ที่กดอยู่ Module ก็จะส่ง Data ออกมาที่ขั้วต่อ 232 จำนวน 3 Byte คือ ASCII 'R' จะถูกส่งออกมาเป็น Byte แรก เพื่อบอกสถานะ การปล่อย Key และ ASCII Code ของ Key ที่ปล่อยจะถูกส่งออกมาเป็น Byte ที่ 2 และจบด้วย 0x0D เป็น Byte สุดท้าย

	ASCII CODE		Hex
	Status Key (Byte 1)	Key Code (Byte 2)	End Byte (Byte 3)
RELEASE KEY	'R' (0x52)	'0-9', 'A-F'	0x0D

Ex. ปล่อย Key 0 ค่าที่ได้ก็จะเป็น
R0 และ 0x0D

ตารางรูปแบบการส่ง data เมื่อมีการปล่อย Key

สรุปการอ่านค่า Key Code แบบ ASCII : จากสถานะการกดและปล่อย Key ที่กล่าวไปข้างต้นในการเขียนโปรแกรมนั้น ผู้ใช้สามารถตรวจสอบสถานะ การกด/ปล่อย Key ได้จากตัวอักษร 'P'=กด Key และ 'R'= ปล่อย Key ซึ่งจะถูกส่งออกมา เป็น Byte แรก และสามารถอ่านค่า Key Code ของ Key ที่ถูกกดหรือปล่อยได้ใน Byte ที่ 2

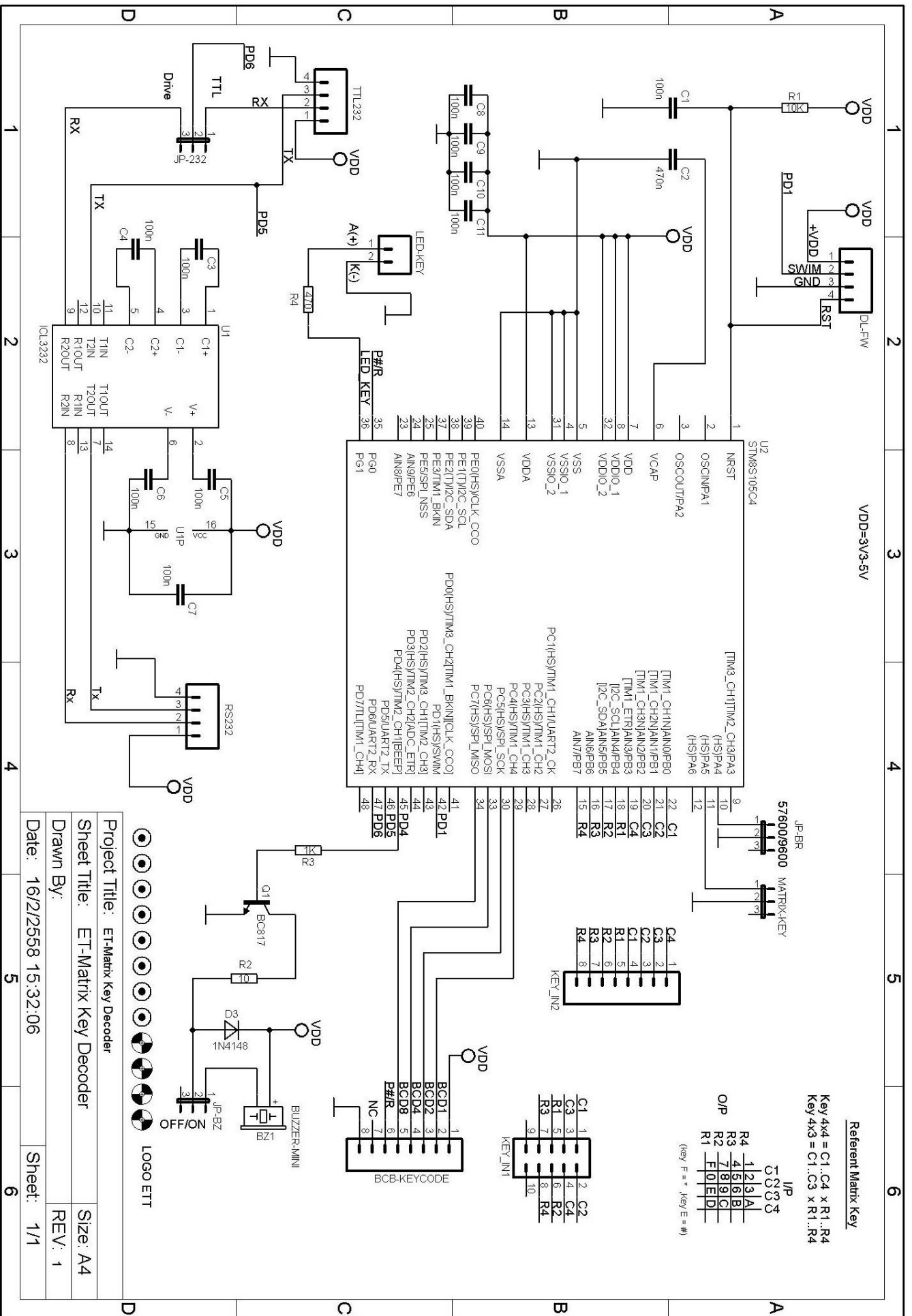
4. ตัวอย่างโปรแกรม

สำหรับตัวอย่างโปรแกรมนั้นจะอยู่ใน CD โดยจะเขียนไว้กับ MCU MCS51 RE2 , PIC18F8722 และ AVR EASY 168(Arduino) ด้วยภาษาซี ซึ่งตัวอย่างในแต่ละ MCU นั้นจะเหมือนกัน ในการทดสอบโปรแกรมตัวอย่างผู้ใช้สามารถ ดูรายละเอียด ในการต่อ Port ทดสอบรวมทั้งรุ่นบอร์ดที่ใช้ทดสอบ และรุ่น Compiler ที่ใช้ในการเขียนโปรแกรมของแต่ละ MCU ได้ที่ Comment ด้านบนของหัวโปรแกรม โดยการทำงานของโปรแกรมสามารถอธิบายได้ดังนี้

Ex1_Read_BCD_Code : สำหรับโปรแกรมนี้อจะเป็นตัวอย่างการอ่านค่า Key Code แบบ BCD 8421 จากตัว Module ไปใช้งาน โดยเมื่อผู้ใช้ต่อ Key pad เข้าที่ Input ของ Module และจ่ายไฟเลี้ยงให้กับ Module แล้ว เริ่มต้นโปรแกรมก็จะทำการตรวจสอบสถานะการกด Key ด้วยการตรวจสอบสถานะของ Pin P/R ถ้าเป็น 0 แสดงว่ามีกรกด Key โปรแกรมก็จะไปทำการอ่านค่า Key Code ที่ได้จากการกด มาเก็บไว้ และส่งค่า Key Code ที่ได้ไปแสดงที่ LED0-LED3 และส่งออกไปยัง Port RS232 (BR=9600) เพื่อแสดงค่า Key Code บน Hyper Terminal ของ PC เช่น ถ้ากด Key 9 ค่าที่แสดงบน LED และ Hyper terminal ก็จะเป็น 0x09 เป็นต้น

Ex2_Read_ASCII_Code : สำหรับโปรแกรมนี้อจะเป็นตัวอย่างการอ่านค่า Key Code แบบ ASCII ผ่านทาง RS232 ของตัว Module ไปใช้งาน โดยจะ Set Baud Rate ในการสื่อสารไว้ที่ 57600 bit/s เมื่อผู้ใช้ต่อ Key pad เข้าที่ Input ของ Module และจ่ายไฟเลี้ยงให้กับ Module แล้ว เริ่มต้นโปรแกรมก็จะทำการตรวจสอบ Data ที่ส่งเข้ามาทาง RS232 ว่ามีการกด Key หรือไม่ ซึ่งถ้า Data ที่ส่งเข้ามาเป็นตัว P แสดงว่ามีการกด Key และ Data ที่ส่งตามมาก็จะเป็นค่า ASCII Code ของ Key ที่กด โปรแกรมก็จะนำค่า Key Code ที่อ่านได้ไปแสดงผลที่ LED ในรูปของ Ascii Code ของ Key ที่กด เช่น ถ้ากด Key 1 ค่าที่อ่านได้บน LED ก็จะเป็น 0x31 ซึ่งก็คือค่า Ascii ของ Key 1 เป็นต้น

***** END *****



Referent Matrix Key

Key 4x4 = C1..C4 x R1..R4
Key 4x3 = C1..C3 x R1..R4

I/P		O/P	
R4	1	2	3
R4	4	5	6
R3	7	8	9
R2	0	1	2
R1	E	F	0

(key: F = *, key E = #)

Project Title: ET-Matrix Key Decoder
 Sheet Title: ET-Matrix Key Decoder
 Drawn By:
 Date: 16/2/2558 15:32:06
 Size: A4
 REV: 1
 Sheet: 1/1